

Research Article

Open Access

Adrian-Gabriel Chifu* and Florentina Hristea

Feature selection for spectral clustering: to help or not to help spectral clustering when performing sense discrimination for IR?

<https://doi.org/10.1515/comp-2018-0021>

Received October 21, 2018; accepted December 30, 2018

Abstract: Whether or not word sense disambiguation (WSD) can improve information retrieval (IR) results represents a topic that has been intensely debated over the years, with many inconclusive or contradictory conclusions. The most rarely used type of WSD for this task is the unsupervised one, although it has been proven to be beneficial at a large scale. Our study builds on existing research and tries to improve the most recent unsupervised method which is based on spectral clustering. It investigates the possible benefits of “helping” spectral clustering through feature selection when it performs sense discrimination for IR. Results obtained so far, involving large data collections, encourage us to point out the importance of feature selection even in the case of this advanced, state of the art clustering technique that is known for performing its own feature weighting. By suggesting an improvement of what we consider the most promising approach to usage of WSD in IR, and by commenting on its possible extensions, we state that WSD still holds a promise for IR and hope to stimulate continuation of this line of research, perhaps at an even more successful level.

Keywords: word sense discrimination, information retrieval, query disambiguation, spectral clustering

1 Introduction

Intuitively, a richer and more linguistically-aware processing of texts should lead to better retrieval results, in general. Natural language processing (NLP) has been used in IR in different ways over time. However, despite many

such efforts, improvements of IR efficiency have not been proven on a large scale evaluation. We argue that the reason is that NLP techniques should not be applied to each query processing, but only to those queries that actually need it – for instance, the ambiguous ones.

Regardless of the great number of existing disambiguation algorithms, the problem of WSD remains an open one. Specific applications may require specific WSD techniques. In fact, task-dependent WSD is categorized [1] as one of the most important open issues for WSD. It stays that way even nowadays.

In a pure IR approach, it is well known that the performance of IR systems is strongly dependent on queries, with term ambiguity being identified as a major cause of query difficulty [2]. In spite of the fact that IR is generally based on query terms and document term matching, without considering the meaning of terms, namely without “understanding” the query¹, more recent approaches [3–5] have taken into consideration the semantics carried out both by the queries and by the documents. Terms were no longer considered as independent, but rather as closely related. The main assumption of such an approach is that context can improve the performance of IR systems.

Various early studies have discussed the fact that the main obstacle in improving IR results when using WSD techniques is the inefficiency of the existing disambiguation algorithms, a problem which increases in the case of the short chunks of text that represent user queries. Their findings and comments appear to be in favor of the skeptical conclusion drawn in [6, 7], where WSD algorithms are required to display at least 90% accuracy in order to be useful for IR. Such comments, however, refer to the straightforward task of WSD, which is either supervised (thus requiring the existence of annotated/parallel corpora) or knowledge-based (thus using

*Corresponding Author: **Adrian-Gabriel Chifu:** Aix Marseille Université, Université de Toulon, CNRS, LIS, Marseille, France; E-mail: adrian.chifu@univ-amu.fr

Florentina Hristea: University of Bucharest, Romania; E-mail: fhristea@fmi.unibuc.ro

¹ Even as late as the year 2006, in a comprehensive book chapter [1], it is argued that the conceptual relationships between words are less important and have less effect on retrieval performance than usage of distributionally-derived representations of documents, for instance.

external knowledge-sources, such as dictionaries or WordNet (WN)-type synsets). The few studies that remain optimistic concerning the possible benefits brought by WSD to IR use mostly supervised WSD techniques [8]. However, it is well known that supervised WSD cannot be used in practice, due to the absence of the necessary annotated/parallel corpora. A majority of studies applying knowledge-based WSD are prone to the same problems [3]: they conduct tests on small data sets [4, 5, 9]; when based on large test collections they report improvements against a weak baseline [6]; the reported improvements are minor ones [10].

Unlike many previous authors, we are suggesting and supporting usage of unsupervised WSD in IR. We consider that a straightforward WSD is not necessary and that unsupervised WSD is the one to be recommended for this type of task, although it has not been extensively used in IR so far. In fact, the most rarely used type of WSD for IR is the unsupervised one, which has been initially recommended by Schütze and Pedersen [11], who report favorable results. It has equally been proven as beneficial in more recent years [3], and from a different perspective, which determines us to focus on it by targeting a specific aspect of the existing discussion – that of whether or not sense discrimination performed by spectral clustering for IR applications can be improved (in this case, by feature selection). Moreover, in the present paper, we will argue that placing the spectral clustering-based disambiguation process at the border between unsupervised and knowledge-based techniques (as a result of performing feature selection) is the most beneficial approach of all.

2 Previous work

Word sense discrimination in IR was introduced by Schütze and Pedersen [11]. The initial approach of these authors proposed a WSD algorithm based on a vector representation of word similarity, derived from lexical co-occurrence, and applied it to the standard vector-space IR model. A different attempt of showing (by means of unsupervised WSD) that the combination of word-based ranking and sense-based ranking is beneficial for IR performance would be made only years later in [12]. This approach involves a method of a completely different nature, one that aims at increasing the top level precision for queries containing ambiguous words. Unlike that described in [11, 13], the method proposed in [12] is based on re-ranking, and not on modifying document representation. It uses, as clustering technique, the classical Naïve

Bayes model, for which semantic WN-based feature selection is performed. In spite of having used feature selection for the Naïve Bayes model, Chifu and Ionescu [12] were not able to overcome the baseline when considering all queries and have therefore targeted the subset of queries with the lowest precision only. The re-ranking method proposed in [12] can be amended with reference to two major aspects: the type of clustering technique to be used for unsupervised WSD and the type of knowledge (feature selection) that could be provided to the clustering algorithm when performing sense discrimination for IR. Subsequent methods of investigation concerning a technique of this type focus their attention [3] on a comparison, in IR, between a classical clustering method, that is aided by feature selection, and a state of the art one that performs its own feature weighting. In [3] sense discrimination is achieved by spectral clustering [14–16] performing its own feature weighting. This approach makes the benefits of using a state of the art clustering method [3] versus a classical one [12] appear as obvious.

Our present study goes beyond the results discussed in [3] by investigating the possible benefits of “helping” spectral clustering as well, through feature selection, when it performs sense discrimination for IR. The types of features we are considering, in order to create a more linguistically informed semantic space for WSD, are of semantic nature (WN-based), of syntactic nature (dependency-based), as well as their combination. Since spectral clustering is known to work best in the presence of the highest number of features [14, 15], the considered features are added to those already selected by spectral clustering itself and their impact on retrieval results, corresponding to large test collections, is observed.

3 The re-ranking method

Chifu *et al.* [3] proposed a re-ranking method for IR that uses unsupervised WSD based on spectral clustering (performing its own feature weighting). They achieved significant improvement of high precision in the case of ambiguous queries (8% above current state of the art baselines) when testing on data collections from the TREC competition². The ambiguity of a query was evaluated with refer-

² TREC7, TREC8, WT10G; queries from this data set contain from 0 to 4 ambiguous terms; baseline: the Terrier search engine with customized parameters.

ence to the title part of the topic³. These authors were interested in improving the level of precision after 5, 10 and 30 retrieved documents (P@5, P@10 and P@30, respectively). Their method is summarized by Algorithm 1.

Algorithm 1: Re-ranking method based on unsupervised WSD

Input: TREC topics, document collection

1. Consider the title part of a topic and check for ambiguous terms.
2. Select the queries that contain at least one ambiguous term.
3. Retrieve the set of documents for each ambiguous query.
4. Add context to queries (descriptive and narrative topic parts).
5. Add contextualized queries to the retrieved document set.
6. Obtain document clusters, by performing WS discrimination for each query and for each ambiguous term using spectral clustering.
7. For each query term, select the document cluster where the query itself was assigned.
8. If there are more than one ambiguous term per query, combine the document clusters selected at step 7 for each individual term.
9. Re-rank initially retrieved document lists by giving more importance to documents that are in the query cluster.

Output: Re-ranked document lists for each query.

Implementation of spectral clustering in step 6 of Algorithm 1 was performed as in [16], where spectral clustering was used for the first time in unsupervised WSD. Namely, the mutual k-nearest-neighbor graph method [14, 17] was chosen for defining the similarity graph that reflects local behavior, starting from a similarity matrix. In the case of this method, the parameter k, representing the number of nearest neighbors, must be set. Chifu *et al.* [3]

³ A query was defined as ambiguous if it contained at least one ambiguous term, with such terms being detected using the WN semantic network. The existence of multiple occurrences of a query word in WN synsets was sufficient for establishing semantic ambiguity. We note that WN synsets were viewed globally and possible lexical ambiguity was not taken into account (no POS-tagger was used), especially in the absence of a context. Therefore, a word's polysemy was determined regardless of its part of speech in the analyzed query.

did not want to fine tune this parameter for each part of speech or each word used in their experiments and have used only one value of k for all the performed tests. According to theoretical considerations [17] it is recommended to use a number of neighbors of the order of the square root of the number of examples (observations). Another important aspect is that of computing the similarity matrix starting from which this similarity graph is defined. In unsupervised WSD the observations are represented by contexts of the ambiguous word. A context is represented as a feature vector and the similarity between two contexts is given by the value of the dot product of the corresponding feature vectors. In [16] the choice of the dot product as measure of similarity is motivated by the success of the linear kernel in supervised WSD [18].

Spectral clustering had also been used, in a similar way, for an unsupervised sense induction task, in [19]. This approach uses the same number of clusters as the number of senses in the Ontonotes sense inventory, in order to study the correspondence between the obtained clusters and the Ontonotes senses. Goyal and Hovy [19] quantitatively and qualitatively analyze their clusters on the Semeval-2010 dataset and comment that this type of clustering should be extremely useful in tasks like Machine Translation and Information Retrieval which require modeling semantics of rare words and unconventional senses of the target words (that may not exist in sense inventories but could be extremely useful in tasks requiring differentiations of senses at various granular levels, as is the case in IR). In the opinion of these authors, identifying the ideal number of clusters to be considered remains the key issue for successfully applying spectral clustering to an IR task.

The present study will be using, in Section 5, the same type of implementation for spectral clustering as in [16], namely a number of neighbors of the order of the square root of the number of examples, in order to enable comparisons with the results obtained by Algorithm 1 that are described in [3].

Let us first note that, since Algorithm 1 re-orders an initially retrieved document list, improvement in recall is not expected. The method targets high precision improvements. It treats the query as if it was another document (by adding it to the set of retrieved documents). The major difference between this method and the one proposed in [12] is usage of spectral clustering in step 6 of Algorithm 1. In order to compare the obtained results with those of Chifu and Ionescu [12], the same fusion functions as in [12] were used. Namely, in step 8 of Algorithm 1, the function CombMNZ [20] is used for fusing the sets of documents corresponding to multiple ambiguous query terms. The re-ranking process organized in step 9 consists of fusing the

initial set of retrieved documents with those obtained as a result of clustering, while taking into account that the documents obtained by the search engine and those obtained after clustering should have different levels of importance in the final results. This is the reason for using a parameter to assign a weight to the fusion function. The function used in [3, 12] has the following structure:

$$S_f^i = S_1^i + \alpha S_2^i$$

with

$$S_1^i = \text{score}(d_i)$$

$$S_2^i = \begin{cases} \text{score}(d_i), & \text{if } d_i \text{ exists in } Clust \\ 0, & \text{otherwise} \end{cases}$$

where S_f^i represents the final score of a document d_i , $\text{score}(d_i)$ represents the score of that document d_i when considered in the initially retrieved document set, $Clust$ is the document cluster containing the query itself and $\alpha \in [0, 1]$ represents the weight of the clustering method for the final results. Testing started with $\alpha = 0$ and increased this parameter by 0.01 at each trial.

In [3] analysis of the results obtained by this method was carried out against the two major approaches existing in the literature, at the time, concerning unsupervised WSD for IR [11, 12]. The obtained results were constantly superior and convincing with respect to sense discrimination being beneficial for IR applications, especially when used from a re-ranking perspective.

The best improvement occurred for the WT10G collection, in the case of P@10. A precision value of 0.2937 was obtained (the baseline was 0.2688), which represents an improvement of 8.48%, statistically significant (paired T-test). This shows a remarkable improvement in high rank precision for ambiguous queries, an important result considering the fact that IR systems are failure-prone in the case of this particular type of queries [6, 21]. And a result which is in favor of using unsupervised WSD in IR.

Chifu *et al.* [3] further refine their analysis by specifically investigating the behaviour of their method over clusters of queries classified by the number of ambiguous terms they contain. They empirically prove that the more ambiguous the query, the better the proposed method works.

This method requires context for queries. Chifu *et al.* [3] create this context using the descriptive and the narrative parts of the involved TREC topics. However, they equally validate the method in the presence of automati-

cally generated context⁴. We can therefore rely on this type of re-ranking for real life applications, a fact which enhances the importance of the proposed technique.

The present work takes the next step and improves the described method by performing feature selection for spectral clustering, as noted in our Introduction, in step 6 of Algorithm 1. Tests have so far been conducted using the WT10G and GOV2 collections (including for creating the disambiguation context).

4 Feature selection

Concerning semantic features, WN-based feature selection was performed by us as introduced in [24]. Namely the feature set was formed by words occurring in the same WN synsets as the target, together with words occurring in related WN synsets and all content words of all corresponding glosses, examples included. The involved semantic relations, that denote the related WN synsets, were chosen according to the part of speech of the target word (after POS-tagging), with all WN senses of the target, corresponding to its part of speech, being considered. Since spectral clustering is known to work best in the presence of the highest number of features [14, 15], we have taken into account a greater number of semantic relations, per part of speech, than recommended by the literature in the case of other tasks [24, 25]. We have thus formed an "extended disambiguation vocabulary", generated by WN, corresponding to each part of speech. In the case of nouns features have been selected using the following WN relations: hyponymy, hypernymy, meronymy, holonymy. For adjectives, the following WN relations have been considered: similarity, also-see, attribute, pertaining-to, antonymy. Corresponding to verbs, we have used the following WN relations: hyponymy, hypernymy, the entailment relation, the causal relation. Finally, for adverbs we have considered antonyms, pertainyms and topics.

Concerning syntactic features, they are provided by dependency relations extracted with Spacy⁵. Features are represented by words that participate in the considered dependencies. At this stage of the study, we have made no qualitative distinction between the different relations, by not taking into account the type of the involved dependencies. Let us note that, within the considered dependen-

⁴ For the creation of which a straightforward pseudo-relevance feedback (PRF) approach [22, 23] is used.

⁵ Spacy performs labeled dependency parsing, is extremely fast and excels at large-scale information extraction tasks: <https://spacy.io/>.

cies, the target word can be either the head or a dependent. The first performed experiment takes into account all undirected first order dependencies anchored at the target word. All words participating in these dependencies (with the exception of the target) will represent the considered syntactic features. The second performed experiment views the target word as head. It takes into account all head-driven dependencies of first order anchored at the target and collects all corresponding dependents which represent the considered syntactic features⁶. The third performed experiment moves to second-order dependencies and (indirectly) enhances the head role of the target (by collecting dependents of the target's dependents, as well), while increasing the number of features⁷.

The types of features we have considered for spectral clustering are the above described ones of semantic nature, those of syntactic nature, as well as their combination. Again since spectral clustering is known to work best in the presence of the highest number of features [14, 15], contrary to the approach in [16], where only features resulted from feature selection were considered, these features of specific types were added to those already selected by spectral clustering itself. Their impact on retrieval results, corresponding to large test collections, was observed. As will be seen (Section 6), test results have suggested that appropriate feature selection is useful even to an advanced, state of the art technique, such as spectral clustering – when used for WSD in IR. Let us also note that the performed feature selection places the spectral clustering-based disambiguation process at the border between unsupervised and knowledge-based techniques.

5 Tests

We have conducted evaluation with reference to the following test collections: WT10G⁸, corresponding to which Chifu *et al.* [3] obtain the best results when applying their method, and GOV2⁹. We have focused on the three top lev-

els of precision studied and improved in [3]: P@5, P@10 and P@30. The usual, standard data preprocessing was performed and the Stanford POS Tagger was used. When implementing Algorithm 1, the retrieval model (step 3) was the language model provided by the Indri search engine, with Dirichlet smoothing ($\mu = 1000$) and with a maximum of 1000 retrieved documents per query. This is a widely used baseline. The same fusion function CombMNZ [20] as in [3] was used (step 8), in order to enable comparisons with the original method. The parameter which assigns a weight to the fusion function (alpha of Figure 1 and Figure 2), representing the weight of the clustering method for the final results, has values ranging from 0 to 2, which are increased by 0.01 at each trial (step 9). In [3] the alpha parameter has a maximum value of 1, but here we want to capture the behavior when the clustering results have even more impact. In the graphs corresponding to the performed experiments the following notations are used: **Baseline** (the run unaltered by the importance of cluster document scores); **Classic** (Algorithm 1 i.e. no feature selection¹⁰); **Classic + semantic** (semantic features are added¹¹); **Classic + syntactic** (syntactic features are added¹²); **Classic + syntactic_head** (syntactic head-driven features are added¹³); **Classic + semantic + syntactic** (semantic and syntactic features are added); **Classic + semantic + syntactic_head** (semantic and syntactic head-driven features are added); **Classic + syntactic_head_2** (first and second-order syntactic head-driven features are added¹⁴).

6 Discussion

In three out of the six studied cases (see Figure 1 and Figure 2), **Classic** provides the worst results (P@30 for WT10G, P@5 and P@10 for GOV2), while for the other three studied cases at least two of the obtained results involving feature selection are superior to the **Classic** ones. Overall, feature selection appears as necessary.

Concerning **P@5**, corresponding to the WT10G collection, semantic WN-based features (in spite of their much lower number) turn out to be more relevant for spectral clustering than dependency-based syntactic ones, with the best result being obtained by **Classic + semantic**. In

⁶ In what follows, we refer to these features as “syntactic head-driven features”.

⁷ Performed test: let us denote the target word by A; collect (as features) all words of type B and C such that B is a dependent of A and C is a dependent of B.

⁸ Approximately 10 GB worth of Web/Blog page documents comprising 1.6 million documents and 100 natural language topic statements.

⁹ Web page collection from the .gov domain, 426 GB (uncompressed), approximately 25 million documents, 150 natural language topic statements.

¹⁰ Features on average: 18513 (WT10G) and 27922 (GOV2).

¹¹ Features on average: 34 (WT10G) and 27 (GOV2).

¹² Features on average: 327 (WT10G) and 685 (GOV2).

¹³ Features on average: 222 (WT10G) and 472 (GOV2).

¹⁴ Features on average: 364 (WT10G) and 732 (GOV2).

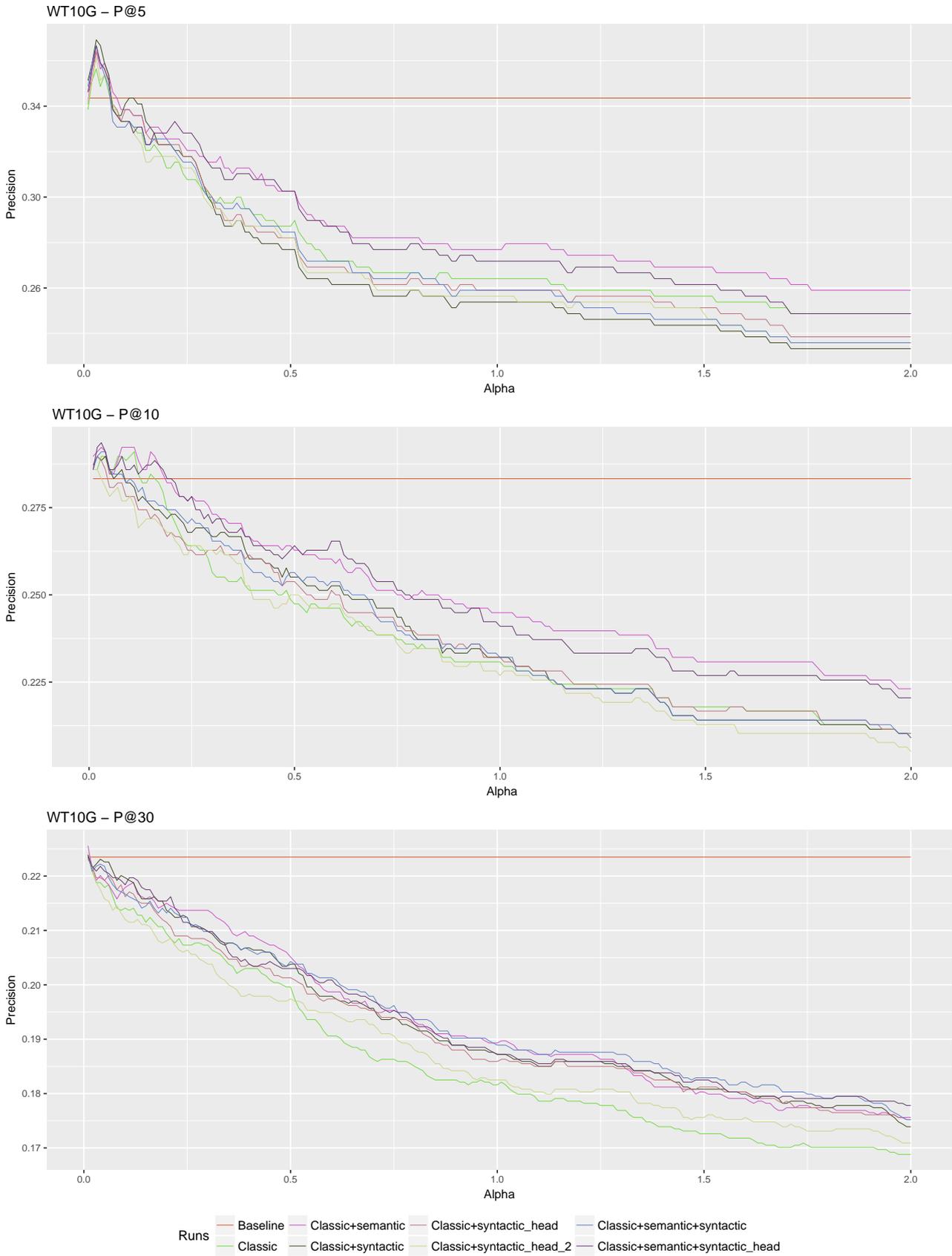


Figure 1: P@X results for WT10G.

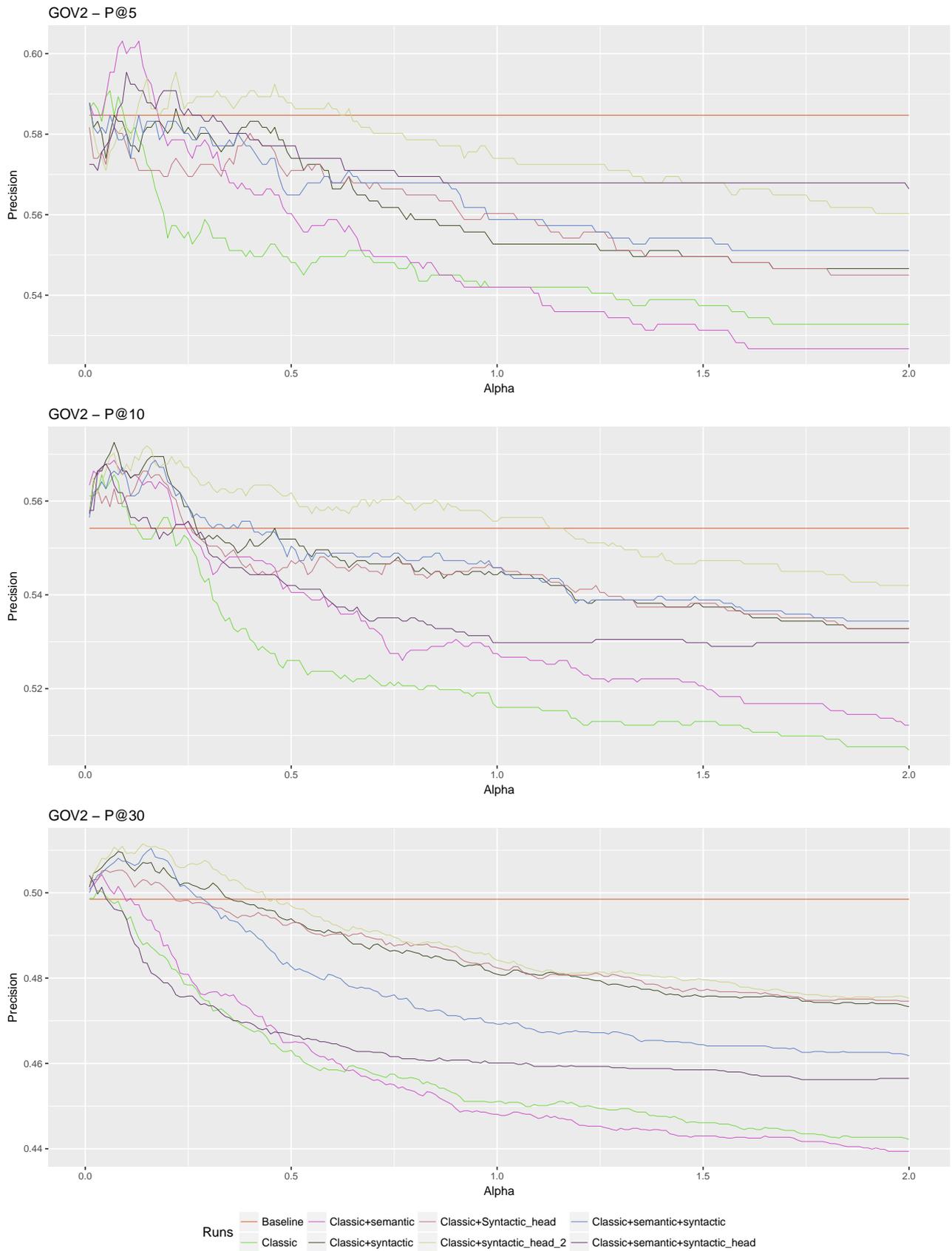


Figure 2: P@X results for GOV2.

Table 1: WT10G. Improvements in percentage with respect to the baseline, for a fixed alpha of 0.05.

| Runs | P@5 | Improvement | P@10 | Improvement | P@30 | Improvement |
|---------------------------------|---------------|--------------|---------------|--------------|--------|-------------|
| Baseline | 0.3436 | - | 0.2833 | - | 0.2235 | - |
| Classic | 0.3538 | 2.97% | 0.2846 | 0.46% | 0.2179 | -2.51% |
| Classic+semantic | 0.3590 | 4.48% | 0.2846 | 0.46% | 0.2192 | -1.92% |
| Classic+syntactic | 0.3590 | 4.48% | 0.2859 | 0.92% | 0.2226 | -0.40% |
| Classic+syntactic_head | 0.3564 | 3.73% | 0.2808 | -0.88% | 0.2192 | -1.92% |
| Classic+syntactic_head_2 | 0.3538 | 2.97% | 0.2782 | -1.80% | 0.2158 | -3.45% |
| Classic+semantic+syntactic | 0.3538 | 2.97% | 0.2859 | 0.92% | 0.2218 | -0.76% |
| Classic+semantic+syntactic_head | 0.3564 | 3.73% | 0.2859 | 0.92% | 0.2209 | -1.16% |

Table 2: GOV2. Improvements in percentage with respect to the baseline, for a fixed alpha of 0.05.

| Runs | P@5 | Improvement | P@10 | Improvement | P@30 | Improvement |
|---------------------------------|---------------|--------------|---------------|--------------|---------------|--------------|
| Baseline | 0.5847 | - | 0.5542 | - | 0.4985 | - |
| Classic | 0.5893 | 0.79% | 0.5626 | 1.52% | 0.4992 | 0.14% |
| Classic+semantic | 0.5893 | 0.79% | 0.5679 | 2.47% | 0.5025 | 0.80% |
| Classic+syntactic | 0.5740 | -1.83% | 0.5687 | 2.62% | 0.5071 | 1.73% |
| Classic+Syntactic_head | 0.5725 | -2.09% | 0.5611 | 1.25% | 0.5053 | 1.36% |
| Classic+syntactic_head_2 | 0.5710 | -2.34% | 0.5664 | 2.20% | 0.5079 | 1.89% |
| Classic+semantic+syntactic | 0.5802 | -0.77% | 0.5626 | 1.52% | 0.5056 | 1.42% |
| Classic+semantic+syntactic_head | 0.5771 | -1.30% | 0.5679 | 2.47% | 0.4987 | 0.04% |

the case of the GOV2 collection, the best result is provided by **Classic + semantic + syntactic_head**. The head role of the target word seems to make a difference (for both the WT10G and the GOV2 collections, **Classic + syntactic_head** results are superior to **Classic + syntactic** ones and **Classic + semantic + syntactic_head** results are superior to **Classic + semantic + syntactic** ones). This is in accordance with the classical Dependency Grammar theory.

Concerning **P@10**, corresponding to the WT10G collection, semantic features again turn out to be superior to syntactic ones, with the best result being obtained by **Classic + semantic**. In the case of the GOV2 collection, the best result seems to be provided by **Classic + semantic + syntactic**, which results in the greatest number of features. The head role of the target does not appear as essential (for the WT10 collection, **Classic + semantic + syntactic_head** is superior to **Classic + semantic + syntactic**; for the GOV2 collection, however, **Classic + semantic + syntactic** is superior to **Classic + semantic + syntactic_head**).

Concerning **P@30**, corresponding to the WT10G collection, the best result seems to be achieved by **Classic + semantic + syntactic**, namely by the greatest number of involved features. In the case of the GOV2 collection, the

best result is obtained by **Classic + syntactic_head**. This is the only experiment in which syntactic features alone lead spectral clustering to the best retrieval result, turning out to be superior to semantic ones, as well as to their combination.

Concerning the importance of the target's head role, results so far appear as inconclusive. While **Classic + syntactic_head** achieves the best result for GOV2, in the case of both test collections, for instance, the combination **Classic + semantic + syntactic** appears as superior to **Classic + semantic + syntactic_head**. Moreover, the differences between **Classic + syntactic** and **Classic + syntactic_head** are not statistically significant (paired T-test) in several cases (**P@10** for WT10G and **P@5** and **P@30** for GOV2). Additionally, tests involving second-order dependencies, which are meant to enhance the head role of the target, also lead to inconclusive results. While in the case of the GOV2 collection, **Classic + syntactic_head_2** seems to represent the best option overall, corresponding to the WT10G collection, these features generate inferior retrieval results, sometimes even compared to **Classic** (see Figure 1 and Figure 2).

To have a clear focus on the improvements, the results obtained for a fixed alpha are displayed in Table 1, for WT10G and in Table 2, for GOV2, respectively. The alpha

parameter is empirically set, for both collections, at 0.05, based on the performance evolution from Figure 1 and Figure 2. The improvements with respect to the baseline are reported as percentage values for all the considered runs. Negative percentages indicate lower performance than the baseline. The best improvements across runs are displayed in bold. One can notice that, whenever improvements occur, the best improvement concerns the feature selection runs. The only exception is the P@5 of GOV2, where **Classic** obtains the highest performance. Even in that case, the best improvement corresponding to **Classic** has the same value as that of **Classic + semantic**, a feature selection run.

7 Conclusion and future study

The present paper answers the question of whether or not it should be attempted to "help" spectral clustering, by means of feature selection, when it performs sense discrimination for IR applications. The question is not a trivial one, considering that this powerful, state of the art clustering method is known as being able to make up for lack of external knowledge of various types, solving many problems on its own, including that of feature selection [16]. However, our insight into feature selection for spectral clustering – when implemented in Algorithm 1 – clearly shows the benefit of selecting features in the case of this task. So far, spectral clustering has been fed knowledge of semantic and/or syntactic type (Section 4). Although this advanced clustering method is known as performing its own feature weighting, corresponding test results acknowledge the utility and even the necessity of performing feature selection for spectral clustering – when used for sense discrimination in IR. This places the spectral clustering-based disambiguation process – performed for IR applications – at the border between unsupervised and knowledge-based techniques.

We consider improving the results obtained by Algorithm 1 as essential, due to its applicability in real life IR scenarios. We, therefore, intend to pursue this line of investigation and to extend it from various perspectives.

Let us once again note that the most rarely used type of WSD for dealing with ambiguous queries is the unsupervised one, although it has been proven to be beneficial at a large scale. The re-ranking method based on spectral clustering introduced in [3] probably represents the most recent important contribution concerning usage of unsupervised WSD in IR, while other, more recent, work in the field [4, 5] continues to make use especially of knowledge-based

WSD. Chifu *et al.* [3] prove that the type of WSD which is used in IR (unsupervised) is of the essence, as well as the involved clustering technique. We hereby further improve retrieval results by placing the spectral clustering-based disambiguation process at the border between unsupervised and knowledge-based techniques (as a result of performing feature selection). However, a clear recommendation as to the most appropriate type of features to be used cannot yet be made. Several perspectives could thus be considered when carrying further this worthwhile type of approach.

Chifu *et al.* [3] create the disambiguation context for queries using the descriptive and the narrative parts of the involved TREC topics. The additional validation of their method in the presence of automatically generated context [3] increases its importance and reliability. However, the automatically generated context offered by these authors [3] to spectral clustering is not an optimal one and it has not yet been optimized. We consider this a major direction for future study.

In the presence of the appropriate context, namely one specifically created for an IR application (for instance, by using specific query expansion methods), spectral clustering could either continue to perform its own feature weighting, as in [3], or it could be aided by feature selection of various types.

Concerning syntactic features, specifically dependency-based ones, we suggest considering the type of the involved dependencies as well, in order to create a more linguistically informed semantic space for WSD. Testing other collections, including TREC Robust, as well as testing other types of features is equally necessary. Concerning the latter, we suggest starting by considering N-gram features [26], which are of a completely different nature. Specifically, they are based on the intuition that the most frequently occurring words near the target can give a better indication of the sense which is activated.

As a result of conducting the described study, we believe that appropriate feature selection for spectral clustering, within an appropriately generated context, could further improve retrieval results, thus making the discussed method even more valuable in real life IR applications.

References

- [1] Resnik P., WSD in NLP applications, In: Agirre E., Edmonds P. (Eds.), *Word Sense Disambiguation: Algorithms and Applications*, Dordrecht: Springer Netherlands, 2006, 299–337
- [2] Mothe J., Tanguy L., Linguistic features to predict query difficulty – a case study on previous TREC campaigns, ACM Conference on

- research and Development in Information Retrieval, SIGIR, Predicting query difficulty – methods and applications workshop, Salvador de Bahia, Brazil, ACM, 2005, 7–10
- [3] Chifu A.-G., Hristea F., Mothe J., Popescu M., Word sense discrimination in information retrieval: a spectral clustering-based approach, *Information Processing & Management*, 2015, 51(2), 16–31
- [4] Tyar S. M., Than M. M., Sense-based information retrieval system by using Jaccard coefficient based WSD algorithm, In: *Proceedings of 2015 International Conference on Future Computational Technologies, ICFCT'15*, 2015, 197–203
- [5] Matinfar F., Hybrid sense disambiguation in web queries, *Bulletin de la Société Royale des Sciences de Liège*, 2016, 85, 1165–1175
- [6] Stokoe C., Oakes M. P., Tait J., Word sense disambiguation in information retrieval revisited, In: *SIGIR*, ACM, 2003, 159–166
- [7] Guyot J., Falquet G., Radhouani S., Benzineb K., Analysis of word sense disambiguation-based information retrieval, In: *Peters C., Deselaers T., Ferro N., Gonzalo J., Jones G. J. F., Kurimo M. (Eds.), CLEF, Lecture Notes in Computer Science*, Springer, 2008, 5706, 146–154
- [8] Zhong Z., Ng H. T., Word sense disambiguation improves information retrieval, In: *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1 (ACL '12)*, Association for Computational Linguistics, Stroudsburg, PA, USA, 2012, 273–282
- [9] Mihalcea R., Moldovan D., Semantic indexing using WordNet senses, In: *Proceedings of the ACL-2000 workshop on Recent Advances in Natural Language Processing and Information Retrieval: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics - Volume 11*, Association for Computational Linguistics, Stroudsburg, PA, USA, 2000, 35–45
- [10] Kim S.-B., Seo H.-C., Rim H.-C., Information retrieval using word senses: root sense tagging approach, In: *Proceedings of the 27th annual international ACM SIGIR conference on Research and Development in Information Retrieval*, ACM, 2004, 258–265
- [11] Schütze H., Pedersen J. O., Information retrieval based on word senses, In: *Proceedings of the 4th annual Symposium on Document Analysis and Information Retrieval*, 1995, 161–175
- [12] Chifu A.-G., Ionescu R.-T., Word sense disambiguation to improve precision for ambiguous queries, *Central European Journal of Computer Science*, 2012, 2(4), 398–411
- [13] Schütze H., Automatic word sense discrimination, *Journal of Computational Linguistics*, 1998, 24(1), 97–123
- [14] Luxburg U., A tutorial on spectral clustering, *Statistics and Computing*, 2007, 17(4), 395–416
- [15] Hastie T., Tibshirani R., Friedman J., *The Elements of Statistical Learning: Data Mining, Inference and Prediction* (2nd edition), New York, USA: Springer-Verlag, 2009
- [16] Popescu M., Hristea F., State of the art versus classical clustering for unsupervised word sense disambiguation, *Artificial Intelligence Review*, 2011, 35(3), 241–264
- [17] Maier M., Hein M., Luxburg U., Optimal construction of k-nearest-neighbor graphs for identifying noisy clusters, *Theoretical Computer Science*, 2009, 410(19), 1749–1764
- [18] Màrquez L., Escudero G., Martínez D., Rigau G., Supervised corpus-based methods for WSD, In: *Agirre E., Edmonds P. (Eds.), Word Sense Disambiguation, Text, Speech and Language Technology*, Springer, Dordrecht, 2007, 33, 167–216
- [19] Goyal K., Hovy E. H., Unsupervised word sense induction using distributional statistics, In: *Hajic J., Tsujii J. (Eds.), Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*, Technical Papers, ACL, 2014, 1302–1310
- [20] Shaw J. A., Fox E. A., Combination of multiple searches, In: *Overview of the 3rd Text Retrieval Conference*, 1995, 105–108
- [21] Mothe J., Tanguy L., Linguistic analysis of users' queries: towards an adaptive information retrieval system, In: *International Conference on Signal Image Technology and Internet-based Systems (SITIS)*, South-East European Research Center (SEERC), 2007, 77–84
- [22] Attar R., Fraenkel A. S., Local feedback in full-text retrieval systems, *Journal of the ACM*, 1977, 24(3), 397–417
- [23] Buckley C., Salton G., Allan J., Singhal A., Automatic query expansion using SMART: TREC 3, In: *Proceedings of The third Text REtrieval Conference (TREC-3)*, 1994, 69–80
- [24] Hristea F., Popescu M., Dumitrescu M., Performing word sense disambiguation at the border between unsupervised and knowledge-based techniques, *Artificial Intelligence Review*, 2008, 30(1), 67–86
- [25] Banerjee S., Pedersen T., Extended gloss overlaps as a measure of semantic relatedness, In: *Proceedings of the 18th International Joint Conference On Artificial Intelligence*, 2003, 805–810
- [26] Preoțiuc-Pietro D., Hristea F., Unsupervised word sense disambiguation with n-gram features, *Artificial Intelligence Review*, 2014, 41(2), 241–260